

Proceedings of Cloud based International Conference ,
 “Computational Systems for Health and Sustainability(CSFHS)”
 5th June, 2020 Organized by sbyetotechnologies.com

A Study On Non-Invasive Blood Glucose Meter Providing Glucose Measurements Painlessly, Without A Blood Sample Or Finger Pricks

UJJAS SAHA

Final Year Student , Dayananda Sagar College of Engineering, Bangalore

Email : ujjassaha12@gmail.com

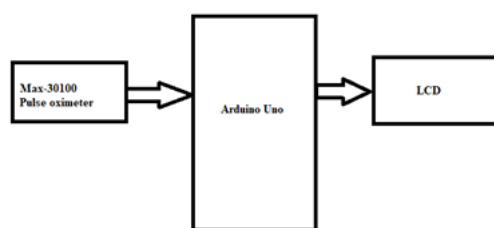
Abstract: Glucose is a simple sugar molecule. The sugar molecule is chemically symbolized as $C_6H_{12}O_6$. This means that glucose molecule contains 6 Carbon (C) atoms, 12 Hydrogen (H) atoms, and 6 Oxygen (O) atoms. In human's blood, glucose molecule circulates as blood sugar. Normally after eating food or drinking, our body breaks down sugars from food and uses them for energy in our cells. To perform this, our pancreas produces a hormone called insulin. Insulin pulls sugar from the blood and puts it in the cells for use. If anyone has diabetes, our pancreas can't produce sufficient insulin. For this, the blood glucose level increases. As a result, our cells fall into much-needed energy shortage. This can lead one to many potential complications including blindness, kidney disease, nerve damage, amputation, stroke, heart attack, and damage to blood vessels etc. Diabetes cannot be cured but it is possible to prevent or control it by keeping glucose level at normal range. Considering this, it is important to regularly check blood glucose level by glucometer.

Key words : Glucose; Sugar Molecule; Kidney; Nerve; Ampulation; Glucometer;

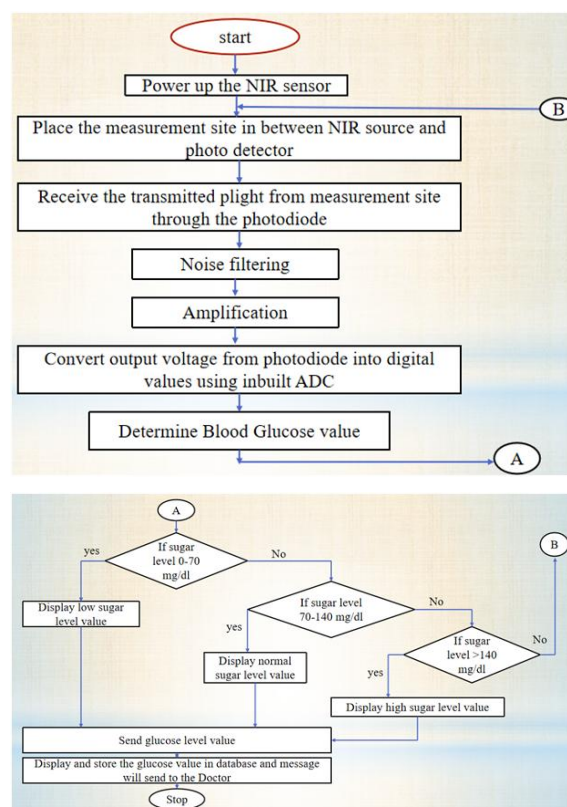
INTRODUCTION

There are different types of glucometers available in the market. But these are invasive. These invasive glucometers need a small amount of blood by puncturing a finger using a needle and put on a test strip which shows the glucose level. Sometimes this method discourages patients because finger puncturing is painful, infectious when the same needle is used for multiple patients, and has a higher cost. Due to this, it is necessary to develop a noninvasive method which does not need finger puncturing and costeffective for diabetic patients.

METHODOLOGY



FLOW CHART



Arduino/GenuinoUno is a microcontroller board based on the ATmega328P. It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz quartz crystal, a USB connection, a power jack, an ICSP header and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a

USB cable or power it with an AC-to-DC adapter or battery to get started. You can tinker with your UNO without worrying too much about doing something wrong, worst case scenario you can replace the chip for a few dollars and start over again. "Uno" means one in Italian and was chosen to mark the release of ArduinoSoftware (IDE) 1.0. The Uno board and version 1.0 of ArduinoSoftware (IDE) were the reference versions of Arduino, now evolved to newer releases. The Uno board is the first in a series of USB Arduino boards, and the reference model for the Arduino platform; for an extensive list of current, past or outdated boards see the Arduino index of boards.

Arduino specification:



Microcontroller	ATmega328P
Operating Voltage	5v
Input voltage	7-12v
Input voltage limit	6-20v
Digital I/O Pins	6
Analogue input Pins	6
DC current per I/O pins	20 mA
DC current for 3.3v Pin	50 mA
Flash Memory	Of which 0.5KB is used
SRAM	2 KB
EEPROM	1KB
Clock Speed	16MHz
Length	68.6mm
Width	53.4mm
Weight	25g

Arduino programming:

The Arduino/Genuino Uno can be programmed with the (ArduinoSoftware (IDE)). Select "Arduino/Genuino Uno" from the Tools>Board menu (according to the microcontroller on your board). The ATmega328 on the Arduino/Genuino Uno comes preprogrammed with a boot loader that allows us to upload new code to it without the use of an external hardware programmer. It communicates using the original STK500 protocol (reference, C header files).

We can also bypass the boot loader and program the microcontroller through the ICSP (In-Circuit Serial Programming) header using ArduinoISP or similar. The ATmega16U2/8U2 is loaded with a DFU boot loader, which can be activated by:

On Rev1 boards: connecting the solder jumper on the back of the board (near the map of Italy) and then resealing the 8U2.

On Rev2 or later boards: there is a resistor that pulls the 8U2/16U2 HWB line to ground, making it easier to put into DFU mode.[1]

Differences with other boards:

The Uno differs from all preceding boards in that it does not use the FTDI USB-to-serial driver chip. Instead, it features the ATmega16U2 (ATmega8U2 up to version R2) programmed as a USB-to-serial converter.[1]

Power:

The Arduino/Genuino Uno board can be powered via the USB connection or with an external power supply. The power source is selected automatically. External (non-USB) power can come either from an AC-to-DC adapter (wall-wart) or battery. The adapter can be connected by plugging a 2.1mm center-positive plug into the board's power jack. Leads from a battery can be inserted in the GND and VIN pin headers of the POWER connector. The board can operate on an external supply from 6 to 20 volts. If supplied with less than 7V, however, the 5V pin may supply less than five volts and the board may become unstable. If using more than 12V, the voltage regulator may overheat and damage the board. The recommended range is 7 to 12 volts. The power pins are as follows:

VIN. The input voltage to the Arduino/Genuino board when it's using an external power source (as opposed to 5 volts from the USB connection or other regulated power source). One can supply voltage through this pin, or, if supplying voltage via the power jack, access it through this pin.

5V. This pin outputs a regulated 5V from the regulator on the board. The board can be supplied with power either from the DC power jack (7 - 12V), the USB connector (5V), or the VIN pin of the board (7-12V). Supplying

voltage via the 5V or 3.3V pins bypasses the regulator, and can damage your board. We don't advise it.

3V3. A 3.3 volt supply generated by the on-board regulator. Maximum current draw is 50 mA.

GND. Groundpins.

IOREF. This pin on the Arduino/Genuino board provides the voltage reference with which the microcontroller operates. A properly configured shield can read the IOREF pin voltage and select the appropriate power source or enable voltage translators on the outputs to work with the 5V or 3.3V.[1]

Memory:

The ATmega328 has 32 KB (with 0.5 KB occupied by the bootloader). It also has 2 KB of SRAM and 1 KB of EEPROM (which can be read and written with the EEPROM library).

Input & Output:

Each of the 14 digital pins on the Uno can be used as an input or output, using `pinMode()`, `digitalWrite()`, and `digitalRead()` functions. They operate at 5 volts. Each pin can provide or receive 20 mA as recommended operating condition and has an internal pull-up resistor (disconnected by default) of 20-50k ohm. A maximum of 40mA is the value that must not be exceeded on any I/O pin to avoid permanent damage to the microcontroller. In addition, some pins have specialized functions:

Serial: 0 (RX) and 1 (TX). Used to receive (RX) and transmit (TX) TTL serial data. These pins are connected to the corresponding pins of the ATmega8U2 USB-to-TTL Serial chip.

External Interrupts: 2 and 3. These pins can be configured to trigger an interrupt on a low value, a rising or falling edge, or a change in value. See the `attachInterrupt()` function for details.

PWM: 3, 5, 6, 9, 10, and 11. Provide 8-bit PWM output with the `analogWrite()` function.

SPI: 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK). These pins support SPI communication using the SPI library.

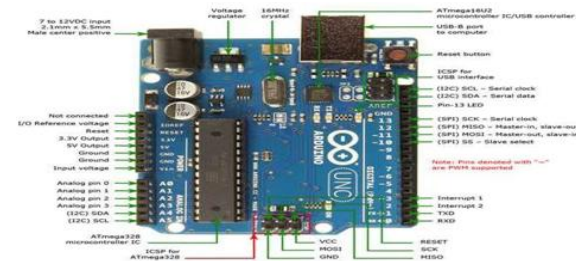
LED: 13. There is a built-in LED driven by digital pin 13. When the pin is HIGH value, the LED is on, when the pin is LOW, it's off.

I2C: A4 or SDA pin and A5 or SCL pin. Support I2C communication using the Wire library.

The Uno has 6 analog inputs, labeled A0 through A5, each of which provide 10 bits of resolution (i.e. 1024 different values). By default they measure from ground to 5 volts, though it is possible to change the upper end of their range using the AREF pin and the `analogReference()` function. There are a couple of other pins on the board:

AREF. Reference voltage for the analog inputs. Used with `analogReference()`.

Reset. Bring this line LOW to reset the microcontroller. Typically used to add a reset button to shields which block the one on the board.[1]



Pin Specification

Communication:

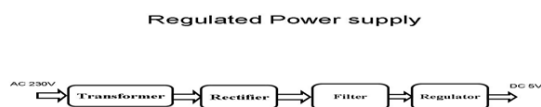
Arduino/Genuino Uno has a number of facilities for communicating with a computer, another Arduino/Genuino board, or other microcontrollers. The ATmega328 provides UART TTL (5V) serial communication, which is available on digital pins 0 (RX) and 1 (TX). An ATmega16U2 on the board channels this serial communication over USB and appears as a virtual COM port to software on the computer. The 16U2 firmware uses the standard USB COM drivers, and no external driver is needed. However, on Windows, an INF file is required. The Arduino Software (IDE) includes a serial monitor which allows simple textual data to be sent to and from the board. The RX and TX LEDs on the board will flash when data is being transmitted via the USB-to-serial chip and USB connection to the computer (but not for serial communication on pins 0 and 1). A Software serial library allows serial communication on any of the Uno's digital pins. The ATmega328 also supports I2C (TWI) and SPI communication. The Arduino Software (IDE) includes a Wire library to simplify use of the I2C bus; see the documentation for details. For SPI communication, use the SPI library.

Automatic (Software) Reset:

Rather than requiring a physical press of the reset button before an upload, the Arduino/Genuino Uno board is designed in a way that allows it to be reset by software running on a connected computer. One of the hardware flow control lines (DTR) of the ATmega8U2/16U2 is connected to the reset line of the ATmega328 via a 100 nanofarad capacitor. When this line is asserted (taken low), the reset line drops long enough to reset the chip. The Arduino Software (IDE) uses this capability to allow you to upload code by simply pressing the upload button in the interface toolbar. This means that the boot loader can have a shorter timeout, as the lowering of DTR can be well-coordinated with the start of the upload. This setup has other implications. When the Uno is connected to either a computer running Mac OS X or Linux, it resets each time a connection is made to it from software (via

USB). For the following half-second or so, the boot loader is running on the Uno. While it is programmed to ignore malformed data (i.e. anything besides an upload of new code), it will intercept the first few bytes of data sent to the board after a connection is opened. If a sketch running on the board receives one-time configuration or other data when it first starts, make sure that the software with which it communicates waits a second after opening the connection and before sending this data. The Uno board contains a trace that can be cut to disable the auto-reset. The pads on either side of the trace can be soldered together to re-enable it. It's labeled "RESET-EN". You may also be able to disable the auto-reset by connecting a 110 ohm resistor from 5V to the reset line.

Regulated power supply:



Transformer:

A transformer is a device that transfers electrical energy from one circuit to another through inductively coupled conductors without changing its frequency. A varying current in the first or primary winding creates a varying magnetic flux in the transformer's core, and thus a varying magnetic field through the secondary winding. This varying magnetic field induces a varying electromotive force (EMF) or "voltage" in the secondary winding. This effect is called mutual induction. If a load is connected to the secondary, an electric current will flow in the secondary winding and electrical energy will be transferred from the primary circuit through the transformer to the load. This field is made up from lines of force and has the same shape as a bar magnet. If the current is increased, the lines of force move outwards from the coil. If the current is reduced, the lines of force move inwards. If another coil is placed adjacent to the first coil then, as the field moves out or in, the moving lines of force will "cut" the turns of the second coil. As it does this, a voltage is induced in the second coil. With the 50 Hz AC mains supply, this will happen 50 times a second. This is called MUTUAL INDUCTION and forms the basis of the transformer.

Rectifier:

A rectifier is an electrical device that converts alternating current (AC) to direct current (DC), a process known as rectification. Rectifiers have many uses including as components of power supplies and as detectors of radio signals. Rectifiers may be made of solid-state diodes, vacuum tube diodes, mercury arc valves, and other components. A device that it can perform the opposite function (converting DC to AC) is known as an inverter. When only one diode is

used to rectify AC (by blocking the negative or positive portion of the waveform), the difference between the term diode and the term rectifier is merely one of usage, i.e., the term rectifier describes a diode that is being used to convert AC to DC. Almost all rectifiers comprise a number of diodes in a specific arrangement for more efficiently converting AC to DC than is possible with only one diode. Before the development of silicon semiconductor rectifiers, vacuum tube diodes and copper (I) oxide or selenium rectifier stacks were used.

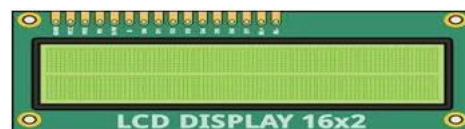
Filter:

The process of converting a pulsating direct current to a pure direct current using filters is called as filtration. Electronic filters are electronic circuits, which perform signal-processing functions, specifically to remove unwanted frequency components from the signal, to enhance wanted ones.

Regulator:

A voltage regulator (also called a 'regulator') with only three terminals appears to be a simple device, but it is in fact a very complex integrated circuit. It converts a varying input voltage into a constant 'regulated' output voltage. Voltage Regulators are available in a variety of outputs like 5V, 6V, 9V, 12V and 15V. The LM78XX series of voltage regulators are designed for positive input. For applications requiring negative input, the LM79XX series is used. Using a pair of 'voltage-divider' resistors can increase the output voltage of a regulator circuit. It is not possible to obtain a voltage lower than the stated rating. You cannot use a 12V regulator to make a 5V power supply. Voltage regulators are very robust. These can withstand over-current draw due to short circuits and also over-heating. In both cases, the regulator will cut off before any damage occurs. The only way to destroy a regulator is to apply reverse voltage to its input. Reverse polarity destroys the regulator almost instantly.

LCD display



16*2 LCD display

A **liquid-crystal display (LCD)** is a flat-panel display or other electronically modulated optical device that uses the light-modulating properties of liquid crystals. Liquid crystals do not emit light directly, instead using a backlight or reflector to produce images in color or monochrome. LCDs are available to display arbitrary images (as in a general-

purpose computer display) or fixed images with low information content, which can be displayed or hidden, such as preset words, digits, and 7-segment displays, as in a digital clock. They use the same basic technology, except that arbitrary images are made up of a large number of small pixels, while other displays have large elements. LCD is used in wide range application including computer monitors, televisions, instrument panels, aircraft cockpit displays, and indoor and outdoor signage. Small LCD screens are common in portable consumer devices such as digital cameras, watches, calculators, and mobile telephones, including smartphones. LCD screens are also used on consumer electronics products such as DVD players, video game devices and clocks. LCD screens have replaced heavy, bulky cathode ray tube (CRT) displays in nearly all applications. LCD screens are available in a wider range of screen sizes than CRT and plasma displays, with LCD screens available in sizes ranging from tiny digital watches to huge, big- screen television sets.

Since LCD screens do not use phosphors, they do not suffer image burn-in when a static image is displayed on a screen for a long time (e.g., the table frame for an aircraft schedule on an indoor sign). LCDs are, however, susceptible to image persistence.

Interfacing an LCD with an Arduino

The 16x2 LCD has a total of 16 pins. As shown in the table below, eight of the pins are data lines (pins 7-14), two are for power and ground (pins 1 and 16), three are used to control the operation of LCD (pins 4-6), and one is used to adjust the LCD screen brightness (pin 3). The remaining two pins (15 and 16) power the backlight. The details of the LCD terminals are as follows:

Terminal 1	GND
Terminal 2	+5V
Terminal 3	Mid terminal of potentiometer (for brightness control)
Terminal 4	Register Select (RS)
Terminal 5	Read/Write (RW)
Terminal 6	Enable (EN)
Terminal 7	DB0
Terminal 8	DB1
Terminal 9	DB2
Terminal 10	DB3
Terminal 11	DB4

Terminal 12	DB5
Terminal 13	DB6
Terminal 14	DB7
Terminal 15	+4.2-5V
Terminal 16	GND

Table . LCD terminals

LCD MODULE



Output of LCD

The name and functions of each pin of the 16x2 LCD module is given below. Pin1 (Vss): Ground pin of the LCD module. Pin2 (Vcc): Power to LCD module (+5V supply is given to this pin) Pin3 (VEE): Contrast adjustment pin. This is done by connecting the ends of a 10K potentiometer to +5V and ground and then connecting the slider pin to the VEE pin. The voltage at the VEE pin defines the contrast. The normal setting is between 0.4 and 0.9V. Pin4(RS): Register select pin. Logic HIGH at RS pin selects data register and logic LOW at RS pin selects command register. If we make the RS pin HIGH and feed an input to the data lines (DB0 to DB7), this input will be treated as data to display on LCD screen. If we make the RS pin LOW and feed an input to the data lines, then this will be treated as a command (a command to be written to LCD controller – like positioning cursor or clear screen or scroll). Pin5 (R/W): Read/Write modes. This pin is used for selecting between read and write modes. Logic HIGH at this pin activates read mode and logic LOW at this pin activates write mode. Pin6 (E): This pin is meant for enabling the LCD module. A HIGH to LOW signal at this pin will enable the module. Pin7 (DB0) to Pin14(DB7): These are data pins. The commands and data are fed to the LCD module through these pins. Pin15 (LED+): Anode of the back light LED. When operated on 5V, a 560 ohm resistor should be connected in series to this pin. In Arduino based projects the back light LED can be powered from the 3.3V source on the Arduino board. Pin16 (LED-): Cathode of the back light LED. RS pin of the LCD module is connected to digital pin 12 of the Arduino. R/W pin of the LCD is grounded. Enable pin of the LCD module is connected to digital pin 11 of the Arduino. This method is very simple, requires less connections and we can almost utilize the full potential of the LCD

module. Digital lines DB4, DB5, DB6 and DB7 are interfaced to digital pins 5, 4, 3 and 2 of the Arduino. The 10K potentiometer is used for adjusting the contrast of the display. The Arduino can be powered through the external power jack provided on the board. +5V required in some other parts of the circuit can be tapped from the 5V source on the Arduino board. The Arduino can be also powered from the PC through the USB port.

JumperWires



A **jump wire** (also known as jumper, jumper wire, jumper cable, DuPont wire, or DuPont cable) is an electrical wire or group of them in a cable with a connector or pin at each end (or sometimes without them – simply "tinned"), which is normally used to interconnect the components of a breadboard or other prototype or test circuit, internally or with other equipment or components, without soldering. Individual jump wires are fitted by inserting their "end connectors" into the slots provided in a breadboard, the header connector of a circuit board, or a piece of test equipment.

MAX30100:

The device has two LEDs, one emitting a red light, another emitting infrared light. For pulse rate, only the infrared light is needed. Both the red light and infrared light is used to measure oxygen levels in the blood. More on that later.



When the heart pumps blood, there is an increase in oxygenated blood as a result of having more blood. As the heart relaxes, the volume of oxygenated blood also decreases. By knowing the time between the increase and decrease of oxygenated blood, the pulse rate is determined. It turns out, oxygenated blood absorbs more infrared light and passes more red light while deoxygenated blood absorbs red light and passes more infrared light. This is the

main function of the MAX30100: it reads the absorption levels for both light sources and stored them in a buffer that can be read via I2C.

Purple MAX30100 Module	Arduino UNO/Nano
VIN	5V
GND	GND
SCL	A5
SDA	A4
INT	D2

The chip has an integrated optical sensor that derives its reading from emitting two wavelength of light from the two LED's then measures the absorbance of pulsing blood through a photodetector. The signal is processed by a low noise analog signal processing unit and communicated to the Microcontroller through the i2C Interface. The MAX30100 operates from 1.8v and 3.3v voltage input and can be powered down through software with negligible standby current, permitting the power supply to remain connected at all times. The device is suitable for wearable devices like smart watch, medical monitoring equipment's, fitness assistant and smart suits. Required Components Arduino Microcontroller, ESP8266 (Arduino IDE Integrated), Teensy MCU (TeensyDuino Integrated), Buzzer / Alarm (Optional) LCD / OLED i2C Display (Optional) Solder Less Breadboard Jumper Wire

Arduino IDE

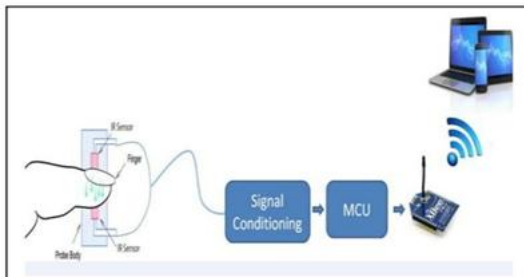
Arduino is a prototype platform (open-source) based on an easy-to-use hardware and software. It consists of a circuit board, which can be programmed (referred to as a microcontroller) and a ready-made software called Arduino IDE (Integrated Development Environment), which is used to write and upload the computer code to the physical board. Arduino provides a standard form factor that breaks the functions of the micro-controller into a more accessible package. A program for Arduino may be written in any programming language for a compiler that produces binary machine code for the target processor. Atmel provides a development environment for their microcontrollers, AVR Studio and the newer Atmel Studio. The Arduino project provides the Arduino integrated development environment (IDE), which is a cross-platform application written in the programming language Java. It originated from the IDE for the languages

Processing and Wiring. It includes a code editor with features such as text cutting and pasting, searching and replacing text, automatic indenting, bracematching, and syntax highlighting, and provides simple one-click mechanisms to compile and upload programs to an Arduino board. It also contains a message area, a text console, a toolbar with buttons for common functions and a hierarchy of operationmenus.

ADVANTAGES

1. Pain Free: No fingerpricking
2. Reading history data in tabular and graphic formats.
3. No disposable: No more waste on test strips and lancets.
4. User Friendly, easy to read data: Easy to operate, clip onto the earlobe.
5. Self-diabetes management easier: Frequent monitoring can help diabetes patients see better and understand their glucose behavior over the time period.
6. Reduced life cycle cost: Onetime expense with virtually unlimited measurements.

APPLICATIONS



Non-Invasive Blood Glucose Level Monitoring System.

CONCLUSION:

In this we have presented a non-invasive blood glucose meter that can provide glucose measurements painlessly, without a blood sample or finger pricks, within a few seconds. The device can be easily adapted to provide continuous blood glucose monitoring and blood oxygen level and maintain a history of these measurements. The device algorithm can also be modified to provide other capabilities like heart rate using the same devices and sensors.

ACKNOWLEDGMENT

The author thanks to Dr. S., Sridhar, Ex Vice Chancellor currently CEO, Sbyte Technologies Chennai, for organizing a cloud-based international

conference and allowing to exhibit our understandings of the work.

REFERENCES

- [1] Tura, A. Maran, and G. Pacini, “Non-invasive glucose monitoring: Assessment of technologies and devices according to quantitative criterion,” Elsevier J. of Diabetes Research and Clinical Practice, vol. 77, no. 6, pp. 16-40, 2007.
- [2] Getting medical devices onto the LAN
- [3] Independa to Integrate Telcare Glucose Monitoring Into Remote Telehealth Suite for Elderly
- [4] Product how-to: Heart rate monitor using a programmable SoC
- [5] Pulse oximetry benefits from the latest programmable SoCs
- [6] V. A. Saptari, “A Spectroscopic system for Near Infrared Glucose Measurement,” PhD Thesis, MIT, 2004.